task 2: Firefighting robot

- Assume that:
- We have a fire fighting robot.
- It keeps moving until **power off switch** is used to turn it off. It uses **three "Ultrasonic Ranging Module HC -SR04"**to check front, right and left paths. If it detects any obstacles in front, it checks right if it is clear, it turns right. Otherwise, it turns left. Add **voltage source for each ultrasonic sensor** to simulate the distance. It uses **two stepper motors** to move forward, turn right or left. If both motors ON, it keeps moving forward. Stopping one motor and moving one side there are obstacles in front . if three has obstacles move backward

[Hint, you can check the ultrasonic and stepper motor examples attached with *SimulIDE*]

- The robot searches for fire. It uses Digital fire detector sensor (Replaced with a **switch**). If the sensor detects fire, it sends HIGH for half a second then returns back to LOW. If it is HIGH for less than half a second then it is not fire and the fighting system should not work. It is just sensor hazard.
- This robot has a fire fighting fan to vanquish fire (appears as a **stepper motor)**. If the robot detects fire it must stop moving and the fan motor should keep rotating until receiving low by 0.25 sec
- The robot has a blinking **led** that blinks every one second.

Note: required components are

- 1 LED: Keep blinking
- 2 switches: power off switch **-** Digital fire detector sensor
- 3 stepper motors: two stepper motors for moving and one as the fan motor
- 3 Ultrasonic modules: front, right and left
- 3 voltage sources: one for each ultrasonic module
- Any number of resistors, if you need.

This task is graded out of 9 based on the following points (one mark for each point)

1. detect obstacles even it appears suddenly
2. make a correct rotation with no delay
3. detect fire successfully even it appears suddenly
4. start firefighting (Fan) correctly
5. stop firefighting (Fan) correctly
6. avoid false fire detecting (HIGH for less than half a second)
7. LED keeps blinking without any change of timing
8. You shouldn't use delay but can delayMicroseconds()
9. You should use one interrupt